

Optimizing Event-Driven Simulations

C. De Michele*

*Dipartimento di Fisica, "Sapienza" Università di Roma,
P.le Aldo Moro 2, 00185 Roma, Italy*

Abstract

Event-driven molecular dynamics is a valuable tool in condensed and soft matter physics when particles can be modeled as hard objects or more generally if their interaction potential can be modeled in a stepwise fashion. Hard spheres model has been indeed widely used both for computational and theoretical description of physical systems. Recently further developments of computational techniques allow simulations of hard rigid objects of generic shape. In present paper we will present some optimizations for event-driven simulations that offered significant speedup over previous methods. In particular we will describe a generalization of well known linked list method and an improvement on nearest neighbor lists method recently proposed by us.

Keywords: Event-driven Molecular Dynamics, Molecular Liquids, Hard Rigid Bodies, Sticky Spots, Linked Cell List, Nearest Neighbour List, Computer Simulations

PACS: 02.70.Ns, 83.10.Rs, 07.05.Tp

1. Introduction

Systems composed of many particles can be modeled as hard rigid bodies (HRB) if excluded volume interactions are dominant and despite absence of any attraction they exhibit a rich phase diagram especially if their shape is non-spherical [1, 2]. The spherical version of these models has already proven to be quite flexible and have been used to tackle, for example, several

*Tel.: +390649913524; fax: +39064463158.

Email address: cristiano.demichiele@roma1.infn.it (C. De Michele)

biological problems [3–5]. The generalization to non isotropic object will increase even more their flexibility and applicability.

Furthermore also attractive interactions between HRBs, provided that they are short-ranged and/or localized, can be properly modeled employing sticky spots (SS) [6–8].

Several numerical techniques have been developed in the past to perform molecular dynamics simulations of particles interacting with only excluded volume interactions. Dealing with hard bodies, the system is propagated in configuration space from one event to next, giving rise to so-called event-driven molecular dynamics (EDMD). The essence of these EDMD numerical algorithms involves the evaluation of the overlap between different objects [9–12] or, equivalently, their geometrical distance [13].

In this paper we propose two optimizations for EDMD, namely: multiple linked lists (MLL) method and nearest neighbor lists with null spots (SNL). MLLs method proved to be very useful in simulating mixtures of hard spheres with very different sizes, while SNL method offers a significant speedup for simulating particles with more complicated shapes, like hard-ellipsoids (HE)[1] or superquadrics (SQ) [10, 13].

2. Simulating Hard Rigid Bodies with Sticky Spots

HRBs can be simulated calculating their distance and collision contact point and time using Newton-Raphson method to solve a proper set of non linear equations as shown in [13]. In [13] we have also shown how to make use of linked lists (LL) and nearest neighbor lists (NNL) to attain good performance. If HRBs have a pronounced non-spherical shape, i.e. a large aspect ratio, it is mandatory to use NNL in order to minimize number of collision predictions.

The event-calendar has been implemented using an hybrid approach as explained in [14], where bounded priority queue is built on top of a binary tree implemented as in [15]. All operations (insertion, deletion and next event retrieve) with such priority queue have a complexity $O(1)$ with respect to particles number. In order to avoid round-off problems for events very close to each other, we shift forward time origin periodically.

Particles interact through an hard core potential and they may be decorated with spherical sticky spots (SS) that interact via a square well potential [8]. Predicting next possible collision means that we have to take into account both collisions between spots and hard core collisions between HRBs. More

specifically if t_{SW} is the next collision time between spots of two particles A and B and t_{HC} is the next collision time between their hard cores, the next event between A and B will occur at time $t_{next} = \min\{t_{SW}, t_{HC}\}$. All the details for an efficient algorithm to find collision times between spots can be found in [6].

SSs have to be considered in calculating escape time from NNL. If t_{HC}^{bb} is the escape time of the HRB from its bounding box (BB) and t_{SS}^{bb} is the escape time of its SSs, then the escape time t_{BB} of a particle from its BB will be $t_{BB} = \min\{t_{HC}^{bb}, t_{SS}^{bb}\}$. t_{HC}^{bb} can be calculated evaluating the time evolution of the distance between the HRB and its BB, i.e. $d(t) = \min_i\{d_i(t)\}$, where $i = 1 \dots 6$ labels each of the 6 BB planes and $d_i(t)$ is the distance between the given HRB and i -th plane.

If the HRB is decorated with SSs we have also to consider the escape time of SSs from its surrounding BB¹. The escape time can be calculated in this case considering simultaneous evolution of all distances between SSs and the 6 BB planes.

3. A Novel Method to Compute Escape Time

In [13] calculation of the escape time t_{HC}^{bb} of an HRB from its enclosing BB requires the evaluation of collision time between the HRB and its BB. This is computationally quite expensive although much faster than the prediction of the collision time between two HRBs. A possible trick is to consider a polyhedron enclosing the HRB and to evaluate the escape time from its BB (see Fig.1). This polyhedron must be chosen to fit the HRB, i.e. given a certain shape (e.g. a parallelepiped) it should have the smallest possible size in order to enclose completely the HRB. Latter requirement will ensure that the escape time of such polyhedron t_{HC}^{sbb} will be an underestimate of the escape time t_{HC}^{bb} , i.e. $t_{HC}^{sbb} < t_{HC}^{bb}$. Evaluation of t_{HC}^{sbb} consists in calculating the smallest escape time of all vertices of the polyhedron. At this point note that such vertices can be thought as spots of null diameter (NSS) meaning that the same algorithm used for calculating the escape time of SSs can be employed to evaluate the escape time t_{HC}^{sbb} . In Fig. 1 the polyhedron is a parallelepiped that encloses the given cylindrical-like HRB. If shape of particles is more complicated than the one show in Fig. 1 simply a polyhedron with a shape

¹Note that in this case BB must enclose both the HRB and the SSs

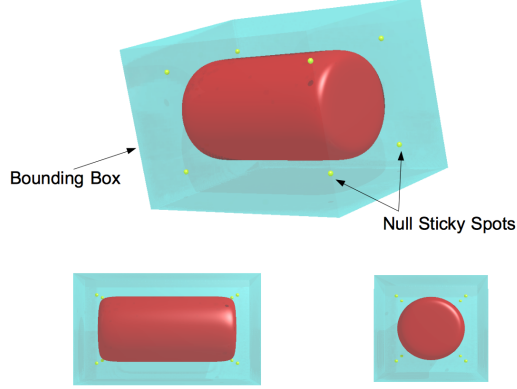


Figure 1: SQ enclosed in its bounding box (cyan parallelepiped) with SS of null diameter, represented here as finite size yellow spheres.

that fits better the HRB shape can be used, i.e. there is no need to calculate analytically any jacobian as in the method proposed in [13].

4. Nearest Neighbour Lists with Null Spots: Performance Results

In this Section we will test performance of the novel method described in Section 3. We consider here superquadrics (SQ), whose surface is defined as follows:

$$f(x, y, z) = |x/a|^n + |y/b|^m + |z/c|^p - 1 = 0 \quad (1)$$

where the parameters n, m, p are real numbers and a, b, c are the SQ semi-axes. A monodisperse system of $N = 512$ SQs has been simulated with $n = 8, m = p = 2$ and with two equal semi-axes, i.e. $b = c$. Such SQs can be characterized by the elongation $X_0 = a/b$ and if elongation $X_0 < 1$ particles are called "oblate", while if $X_0 > 1$ particles are called "prolate". For this test we have taken into account only prolate SQs, whose shape resembles that of a cylinder with smoothed edges (see Fig. 1). In particular elongations $X_0 = 1.0, 2.0, 3.0$ have been studied. The system of prolate SQs has been simulated in a cubic box of volume V with periodic boundary conditions at the volume fractions $\phi = 0.20, 0.30, 0.35, 0.40$ ($\phi = \pi X_0 b^3 \rho / 6$, where $\rho = N/V$ is the number density). The length of the smallest semi-axes is chosen to be the unit of length ($b = 1.0$), the mass of the SQ is the unit of

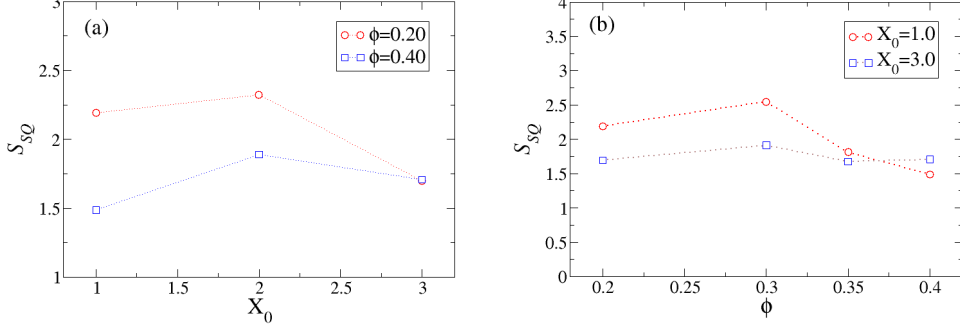


Figure 2: (a) Speedup S_{SQ} versus elongation X_0 for $\phi = 0.20, 0.40$ (b) Speedup S_{SQ} versus ϕ for $X_0 = 1.0, 3.0$.

mass ($m = 1$) and the moment of inertia is spherically symmetric and equal to 1.0. To create the starting configuration at a desired ϕ , an extremely diluted crystal has been melted; afterwards, the particles have been grown independently up to the desired packing fraction (quench in ϕ at fixed N , X_0), similarly to what was done in [13].

To test the algorithm speed, we use the *CPU time*² per collision, i.e. $\tau_c = T_{tot}/N_{coll}$, where T_{tot} is the (real) time needed to perform N_{coll} collisions during a simulation.

We can define the speedup S_{SQ} as follows:

$$S_{SQ} = \tau_c^N / \tau_c^S \quad (2)$$

where N refers to simulations that use NNL and S refers to simulations that make use of SNL. Fig. 2 (a) shows S_{SQ} as a function of elongation X_0 for two different volume fractions and it is apparent that use of SNL offers a speedup around 2 independently of elongation. In a similar way Fig. 2 (b) shows speedup S_{SQ} for two elongations $X_0 = 1.0, 3.0$ as a function of ϕ and it turns out again that S_{SQ} is also independent of ϕ . Hence the novel method proposed for calculating the escape time of a HRB from its BB provides a significant speedup over previous methods and this speedup is quite independent of X_0 and ϕ . This result can be fairly understood

²*CPU time* means the real time spent by the CPU for calculations.

because number of collision predictions for a given X_0 and ϕ are independent of method used to calculate the escape time (either SNL or NNL). For that reason algorithm performance depends mainly on efficiency in rebuilding SNL or NNL, but SNL are more efficient than NNL and *CPU time* needed to calculate escape time is quite insensitive to elongation and volume fraction.

5. Binary Mixtures of Hard-Sphers: Multiple Linked Lists

Linked lists³ (LL) [15] are commonly employed in molecular dynamics simulations in order to avoid to check all the N^2 possible collisions among N simulated particles. In the LL method, the simulation box is partitioned into M^3 cells and only collisions between particles inside the same cell or its 26 adjacent cells are accounted for. This also means that, whenever an object crosses a cell boundary going from cell a to a new cell b , it has to be removed from cell a and added to cell b . Several variants of original LL method have been proposed in literature [16] with a view to improving the original LL method (as discussed for example in [17]). All these improved LL methods are intended to avoid unnecessary distance calculations but they do not tackle the case of a system composed of different species having very different sizes. Here we consider the case of a system composed of N_s several species having different sizes $\{\sigma_1, \dots, \sigma_{N_s}\}$ and for the sake of simplicity we can assume that particles are spheres.

For making use of the original LL method, the simulation box must be partitioned into cubic cells and each cell must have a side length slightly greater than $\sigma_{max} = \max_i\{\sigma_i\}$. In general, this restriction can considerably compromise algorithm performance. To understand this, consider m_i particles of diameter σ_i inside a cell and $q_i = (\sigma_{max}/\sigma_i)^3$, at a fixed volume fraction, if $q_i \rightarrow 0$ then $m_i \rightarrow \infty$ and performance are severely compromised.

A possible generalization of the original LL method for dealing with such polydisperse system consists in using one different LL for each pair of species, meaning that for each pair of species a different partitioning of simulation box into cells is used as illustrated pictorially in Fig. 3 for a binary mixture of hard spheres.

In general if one has N_s different species, $N_s(N_s + 1)/2$ different LLs must be used where the cell side length for interacting species l and m ⁴ has to be

³they are also referred to as linked cell lists

⁴Assuming that they are additive.

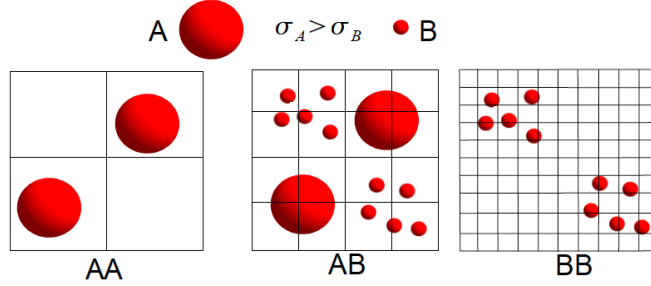


Figure 3: Pictorial representation of Multiple Linked Lists for a binary mixture of two particles where diameter σ_A of particles A is bigger than diameter σ_B of particles B . For each possible interaction AA , AB and BB a different partitioning of simulation box into cells is employed, i.e. multiple linked cell lists are used.

greater than $(\sigma_l + \sigma_m)/2$. Implementation of MLL is quite straightforward except for the following remark: if particles, as they cross simulation box boundaries, are reinserted into the box with periodic boundary conditions, one has to ensure that reinsertion happens only once.

6. Speedup of Multiple Linked Lists

In this Section we will test performance of MLL method for a binary mixture of spheres having very different size. The two species (labeled by A and B) are characterized by a diameter ratio $\sigma_A/\sigma_B = q > 1$ and their masses are chosen to be equal and unitary. N_A and N_B will be number of particles A and B respectively and $N = N_A + N_B$. The number of particles A will be kept fixed to 250, i.e. $N_A = 250$. We will investigate the algorithm performance varying N , the volume fraction ϕ and the size ratio q . The simulation box is cubic with periodic boundary conditions. As for SQs we make use of the $O(1)$ event-calendar proposed in [14].

Again we define the speedup S_{SQ} as follows:

$$S_{BM} = \tau_c^{LL} / \tau_c^{MLL} \quad (3)$$

where LL refers to simulations that use single LL, MLL refers to simulations that employ MLL and τ_c is the *CPU time* per collision. Fig. 4 (a) shows S_{BM} as a function of ϕ for two different size ratios q . It is remarkable that speedup ranges from 15 to 40.

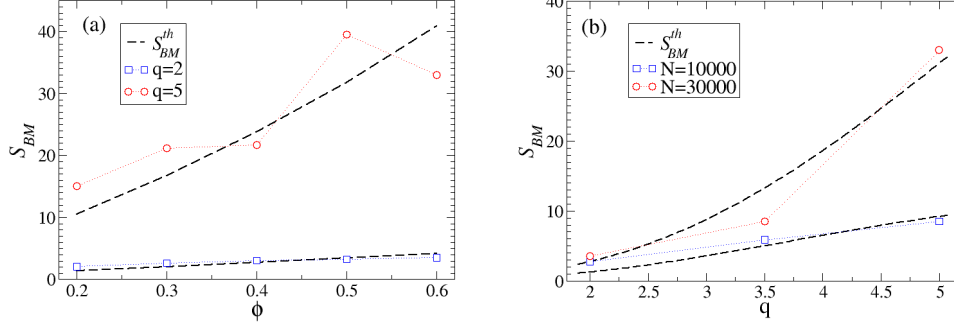


Figure 4: (a) Speedup S_{BM} versus total volume fraction ϕ for binary mixtures for $q = 2, 5$ with $N = 30000$ for all points. (b) Speedup S_{BM} versus size ratio q for $N = 10000, 30000$ with $\phi = 0.60$ for all points. Dashed lines are fits to function S_{BM}^{th} defined in Eq. (5).

The number of spheres of type B , N_B^{cell} , within a cell using conventional LL method is roughly:

$$N_B^{cell} = \frac{6 q^3 \phi}{\pi [q^3(1 - \phi)N_A/N_B + 1]} \quad (4)$$

and a analytical estimate S_{BM}^{th} for the speedup turns out to be ⁵:

$$S_{BM}^{th} = K N_B^{cell} \quad (5)$$

where K is an arbitrary constant. Fig. 4 (a) and (b) shows also fits to simulations data of function S_{BM}^{th} defined in Eq. (5). Agreement between numerical data and S_{BM}^{th} proves that S_{BM}^{th} is a reasonable estimate.

7. Conclusions

In this paper two novel techniques easy to implement have been proposed for optimizing EDMD simulations. The SNL method offers a nearly constant speedup around 2 with respect to the old method proposed in [13] and it can be easily adapted to more complicated shapes of simulated particles. The SNL method is actually used for simulating a recently developed model

⁵for MLL method N_B^{cell} is of the order of 1.

of DNA duplexes (DNAD) [18] consisting in cylindrical-like SQs decorated with two sticky spots on their two bases in order to model stacking interactions between DNADs. The details of latter study will be given in a future publication.

MLLs have been already used in [19] where spherical particles, evolving according to event-driven brownian dynamics [20], can be absorbed by one fixed spherical sink whose size may be much greater than diffusing particles. MLLs are also currently used for investigating phase diagram of a binary mixture of hard spheres whose size ratio $q = 5$ upon changing the partial volume fractions of two species. For this system it is possible to calculate a theoretical phase diagram with respect to glass transition within the framework of Mode Coupling Theory and recently jamming lines for such system have been evaluated both theoretically (using Replica theory) and numerically[21]. In view of growing interest for this system it is crucial to have an efficient algorithm to explore whole phase diagram. It is worth noting that also conventional time-driven molecular dynamics may benefit from MLL. Huge increase in performance provided by MLL, when the length scales between the components is so important, could play a relevant role in multiscale simulations, a topic that is attracting a lot of interest for biological and material applications.

Finally the two optimization methods, SNL and MLL, illustrated in this paper can be used together, like LL and NNL, by using MLL to generate SNL.

Acknowledgments

The author acknowledges support from ERC (226207-PATCHYCOLLOIDS) and warmly thanks Prof. G. Foffi for careful reading of the manuscript.

References

References

- [1] C. De Michele, F. Sciortino, R. Schilling, Dynamics of uniaxial hard ellipsoids, Phys. Rev. Lett. 98 (2007) 265702.
- [2] D. Frenkel, B. M. Mulder, J. P. McTague, Phase diagram of a system of hard ellipsoids, Phys. Rev. Lett. 52 (1984) 287–290.

- [3] D. C. Rapaport, Role of reversibility in viral capsid growth: A paradigm for self-assembly, *Phys. Rev. Letters* 101 (2008) 186101.
- [4] A. Stradner, G. Foffi, N. Dorsaz, G. Thurston, P. Schurtenberger, New insight into cataract formation: Enhanced stability through mutual attraction, *Phys. Rev. Letters* 99 (2007) 198103.
- [5] J. M. Borreguero, B. Urbanc, N. D. Lazo, S. V. Buldyrev, D. B. Teplow, H. E. Stanley, Folding events in the 21 – 30 region of amyloid β -protein ($\alpha\beta$) studied in silico, *Proc. Natl. Acad. Sci.* 102 (2005) 6015–6020.
- [6] C. De Michele, S. Gabrielli, P. Tartaglia, F. Sciortino, Dynamics in the presence of attractive patchy interactions, *J. Phys. Chem. B* 110 (2006) 8064.
- [7] C. De Michele, P. Tartaglia, F. Sciortino, Slow dynamics in a primitive tetrahedral network model, *J. Chem. Phys.* 125 (2006) 204710.
- [8] S. Corezzi, C. De Michele, E. Zaccarelli, D. Fioretto, F. Sciortino, A molecular dynamics study of chemical gelation in a patchy particle model, *Soft Matter* 4 (2008) 1173–1177.
- [9] A. Donev, S. Torquato, F. H. Stillinger, Neighbor list collision-driven molecular dynamics simulation for nonspherical hard particles. ii. applications to ellipses and ellipsoids, *Journal of Computational Physics* 202 (2005) 765–793.
- [10] Y. Jiao, F. H. Stillinger, S. Torquato, Superdisks and the role of symmetry, *PRL* 100 (2008) 245505.
- [11] J. Vieillard-Baron, Phase transitions of the classical hard-ellipse system, *Journal of Chemical Physics* 56 (1972) 4729–4744.
- [12] J. Perram, M. Wertheim, Statistical mechanics of hard ellipsoids. i. overlap algorithm and the contact function, *J. Comp. Phys.* 58 (1985) 409–416.
- [13] C. De Michele, Simulating hard rigid bodies, *J. Comput. Phys.* 229 (2010) 3276–3294.
- [14] G. Paul, A complexity $O(1)$ priority queue for event driven molecular dynamics simulations, *J. Comput. Phys.* 221 (2007) 615–625.

- [15] D. C. Rapaport, *The Art of Molecular Dynamics Simulation*, Cambridge University Press, 2004.
- [16] U. Welling, G. Germano, Efficiency of linked cell algorithms, arXiv:1006.1239v1 (2010) 1–13.
- [17] M. P. Allen, D. J. Tildesley, *Computer simulation of liquids*, Clarendon Press, paperback 385pp edition, 1989.
- [18] M. Nakata, G. Zanchetta, B. D. Chapman, C. D. Jones, J. O. Cross, R. Pindak, T. Bellini, N. A. Clark, End-to-end stacking and liquid crystal condensation of 6to 20base pair dna duplexes, *Science* 318 (2007) 1276–.
- [19] N. Dorsaz, C. De Michele, F. Piazza, P. D. L. Rios, G. Foffi, Diffusion-limited reactions in crowded environments, arXiv:1007.2529, Accepted by *Phys. Rev. Lett.* (2010) 1–4.
- [20] A. Scala, C. De Michele, Th. Voigtmann, Event-driven brownian dynamics for hard spheres, *J. Chem. Phys.* 126 (2007) 134109.
- [21] I. Biazzo, F. Caltagirone, G. Parisi, F. Zamponi, Theory of amorphous packings of binary mixtures of hard spheres, *Phys. Rev. Lett.* 102 (2009) 195701.